Problem 1

Assume that before the host opened one of the doors, the incidents of prize behind each door are A1, A2, A3, and the probability is the same. So we can derive:

$$P(A1) = P(A2) = P(A3) = 1/3$$

Assume the incidents that the host opened each door are B1, B2, B3. If my friend picked door 1 and the host would randomly choose between door 2 and door 3, then we can derive:

$$P(B2|A1) = 1/2,\ P(B2|A2) = 0,\ P(B2|A3) = 1$$

Assume that my friend picked door 1 and the host opened door 2, then the probability of prize behind door 1 is:

$$P(A1|B2) = P(B2|A1)\ P(A1)\ /\ P(B2)$$

According to the general probability rule, we can derive P(B2) by:

$$P(B2) = P(B2|A1)\ P(A1) + P(B2|A2) + P(B2|A3)\ P(A3) = 1/2$$

So the posterior probability is:

$$P(A1|B2) = 1/3$$

According to the fact that there was no prize behind the opened door:

$$P(A1|B2) = 0$$

We can derive:

$$P(A3|B2) = 1 - P(A1|B2) - P(A1|B2) = 2/3$$

It means that after the host opened the door, the probability of prize behind the chosen door is less than the other door. Therefor she should change her original choice.

Problem 2

For Xi ~ Multinominal(π):

$$P(X|\pi) \propto \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{X_{kn}}$$

According to Bayes rules:

$$P(\pi|X) \propto P(X|\pi)\, P(\pi)$$

To make it conjugate, prior should have the same form:

$$P(\pi|\alpha) \propto \prod_{k=1}^{K} \pi_k^{\alpha_k} \ , \quad \alpha = \{\alpha1, \alpha2, \dots, \alpha k\}$$

So the posterior distribution is:

$$P(\pi|Xk) \propto \prod_{k=1}^{K} \pi_k^{\sum_{n=1}^{N} X_{nk} + \alpha_k}$$

1)The name of the posterior distribution is Dirichlet Distribution.

2)The most obvious feature is that we can easily calculate the parameters based on prior parameters and data set.

Problem 3

a)

According to Bayes rule:

$$p(\lambda|X) = \frac{p(X|\lambda)p(\lambda)}{\int p(X|\lambda)p(\lambda)d\lambda}$$

Since Xn ~ i.i.d Poisson(λ) and λ ~ Gamma(a,b), we can derive the posterior:

$$p(\lambda|X) = \frac{(N+b)^{\sum_{n=1}^{N} x_n + a}}{\Gamma(\sum_{n=1}^{N} x_n + a)} \lambda^{(\sum_{n=1}^{N} x_n)+a-1} e^{-(N+b)\lambda}$$

Thus $p(\lambda|X) \sim \text{Gamma}(\sum_{n=1}^{N} x_n + a, N+b)$

b)

By substituting the above outcome:

$$p(x^*|x1, \ldots, xn) \propto \int_0^{\infty} \lambda^{(\sum_{n=1}^{N} x_n)+x^*+a-1} e^{-(1+N+b)\lambda}\, d\lambda$$

By adding the constants:

$$p(x^*|x_1, \ldots, x_n) = \frac{1}{x^*!} \frac{(N+b)^{\sum_{n=1}^{N} x_n + a}}{\Gamma(\sum_{n=1}^{N} x_n + a)} \frac{\Gamma(\sum_{n=1}^{N} x_n + a + x^*)}{(1+N+b)^{\sum_{n=1}^{N} x_n + a + x^*}}$$

Problem 4

a) Matlab code for classification

```
X_train = csvread('X_train.csv');
y_train = csvread('label_train.csv');
y_test = csvread('label_test.csv');
X_test = csvread('X_test.csv');

setNum = size(X_test,1);
setSize = size(X_train,2);
N1 = length(find(y_train));
N0 = length(find(~y_train));
p_pre1 = zeros(setNum,1);

sumX1 = sum(X_train.*repmat(y_train,1,setSize),1);
sumX0 = sum(X_train.*repmat(1-y_train,1,setSize),1);
log_cX = sum((sumX0+1)*(log(N0+1)-log(N0+2))) -
sum((sumX1+1)*(log(N1+1)-log(N1+2)));
log_cN = log(N1+2)-log(N0+2);

for k = 1:setNum
    log_factor1 = 0;
```

```
    log_factor0 = 0;
    for i = 1: 54
        if X_test(k,i) ~= 0
            log_factor1 = log_factor1 +
sum(log(sumX1(i)+1:sumX1(i)+X_test(k,i)));
            log_factor0 = log_factor0 +
sum(log(sumX0(i)+1:sumX0(i)+X_test(k,i)));
        end
    end
    log_fx = (sum(X_test(k,:))-1)*log_cN + log_factor0 - log_factor1;
    p0_div_p1 = exp(log_fx + log_cX);
    p_pre1(k) = 1/(1+p0_div_p1);
end
y = (p_pre1 > 0.5);
```

b) The confusion matrix is:

	classified_spam	classified_non_spam
spam	172	10
non-spam	48	231

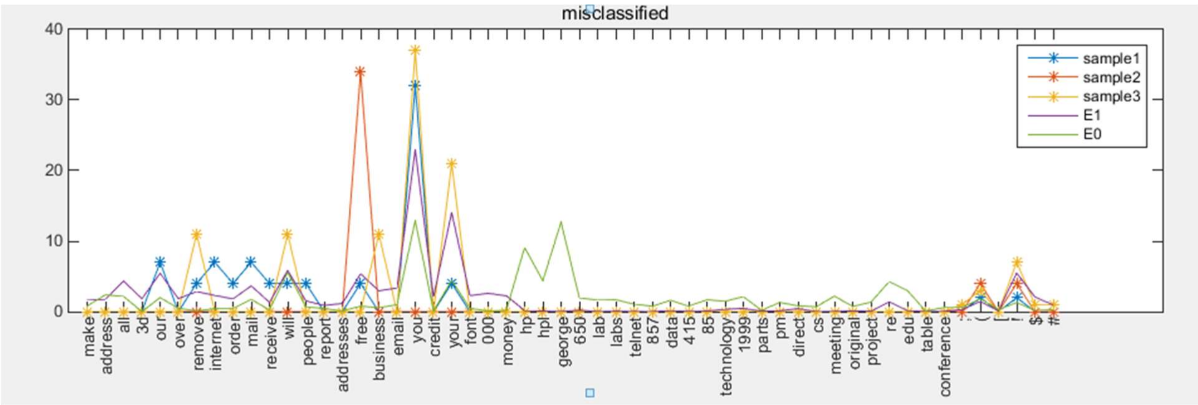
c) The predictive probabilities for three misclassified emails are:

Sample1: P(y=1)=0.0712 ,P(y=0)=0.9288

Sample2: P(y=1)=1.1318e-84 , P(y=0)=0.9999

Sample3: P(y=1)=5.3502e-06 , P(y=0)=0.9999

And the figure is:



d) The predictive probabilities for three most ambiguous emails are:

Sample1: P(y=1)=0.4077 , P(y=0)=0.5923

Sample2: P(y=1)=0.3840 , P(y=0)=0.6160

Sample3: P(y=1)=0.3722 , P(y=1)=0.6378

And the figure is: